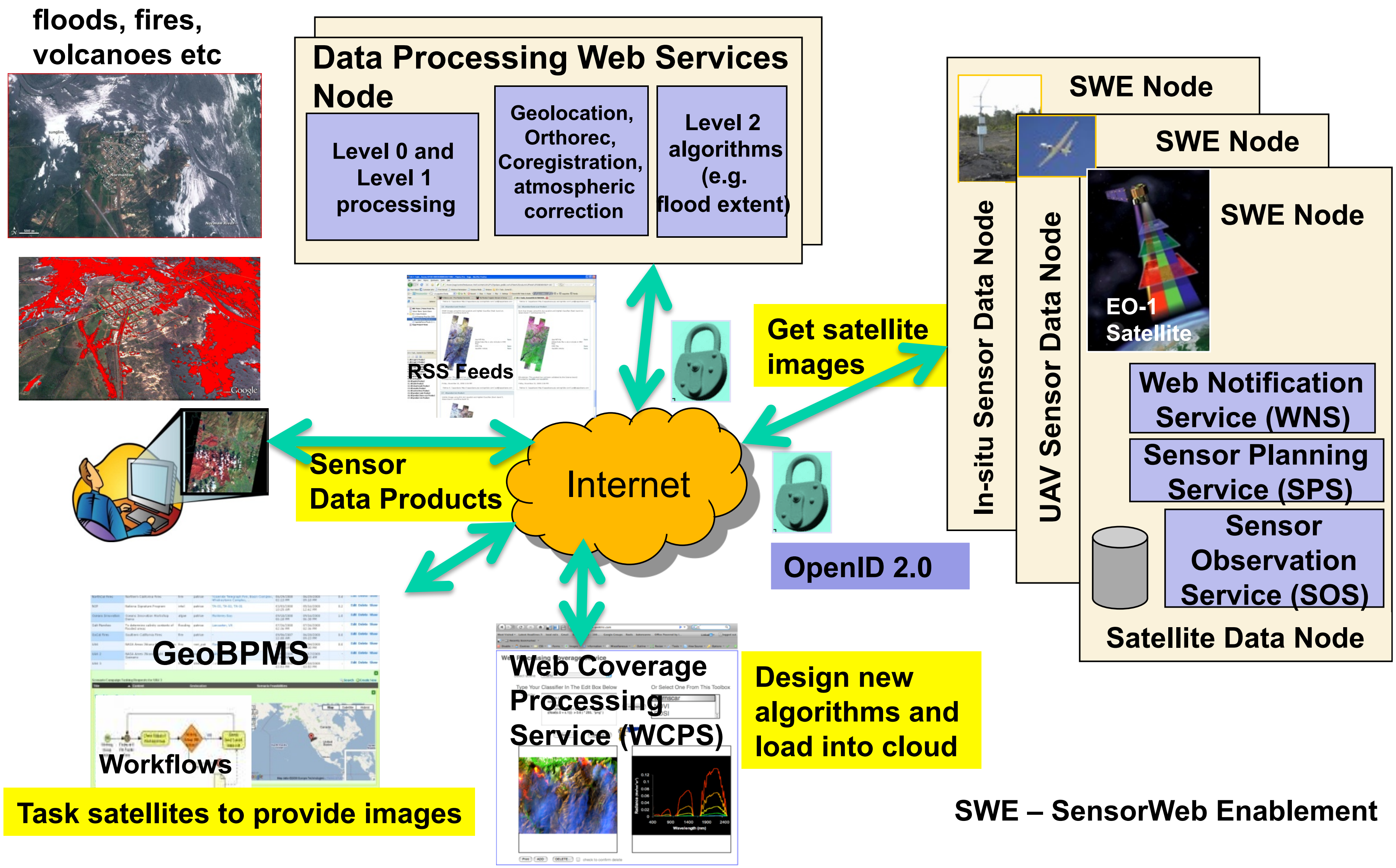
SensorWebs for Easy Access to Satellite Data and Rapid Data Product Delivery

Dan Mandl

IS&T Colloquium September 12, 2012

SensorWeb High Level Architecture

Sensors, Algorithms and Models Wrapped in Web Services Provide Easy Access to Sensor Data and Sensor Data Products



Add Cloud Computing

CREST Hydrological Model

TRMM based Global Rainfall Estimates

Radarsat Images & flood extent maps

MODIS Daily Flood Extent Map

Global Disaster and Alert and Coordination System (GDACS)

Namibian River Gauge Stations - Daily Measurements

Storage – 1 year
Hyperion & ALI Level 1R

Storage – 1 year
Hyperion & ALI Level 1G

Storage – 1 years
Hyperion & ALI Level 1R and Level 1G AC

Storage – 1 year
User Defined L2 Products
e.g. EO-1 Flood Mask

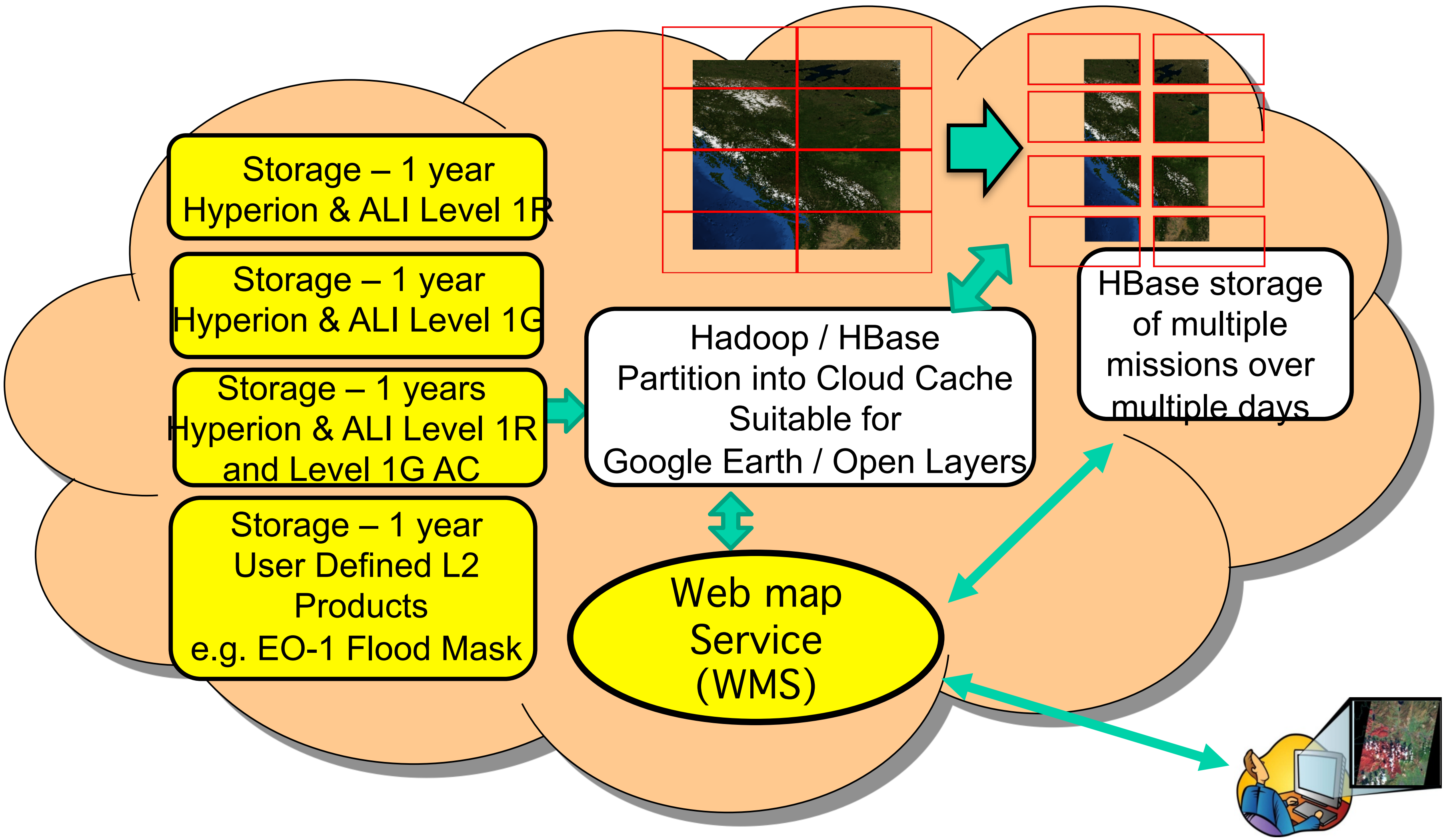
Namibia River Gauge Data base

Flood Dashboard Display Service
- Mashup
- Google Maps Inset
- Plot Package

http server

- **Eucalyptus/Open Stack-based Elastic Cloud SW**
- **300+ core processors**
- **40 x 2 Tbytes of storage**
- **10 Gbps connection to GSFC**
  - **being upgraded to 80 Gbps (Part of OCC)**
- **Hadoop/Tiling**
- **Supplied by Open Cloud Consortium**
- **Open Science Data Cloud Virtual Machines & HTTP server to VM's**

Hadoop and Tiling Handles Large Dataset Displays



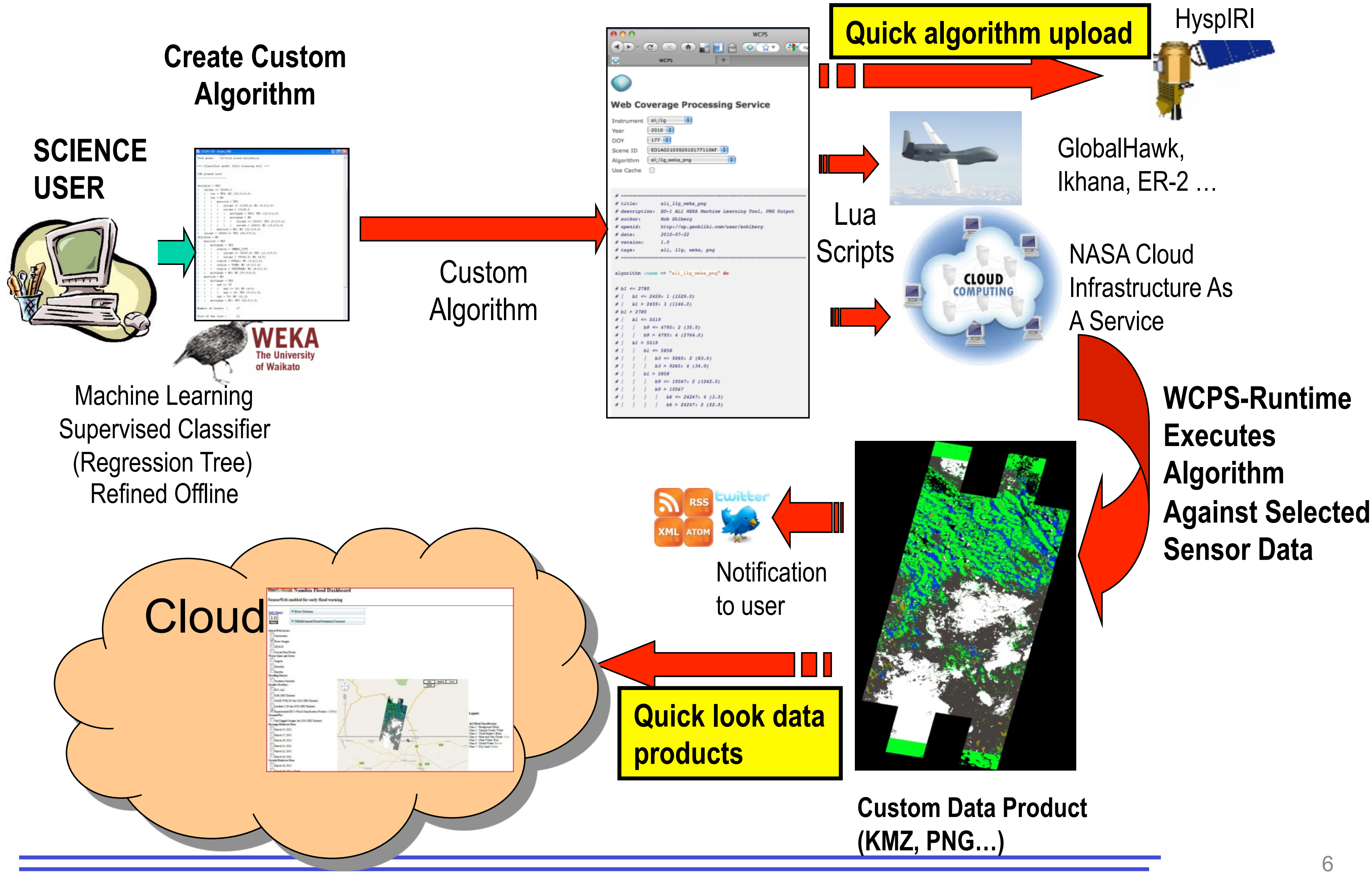
Flooding in Divundu, Namibia

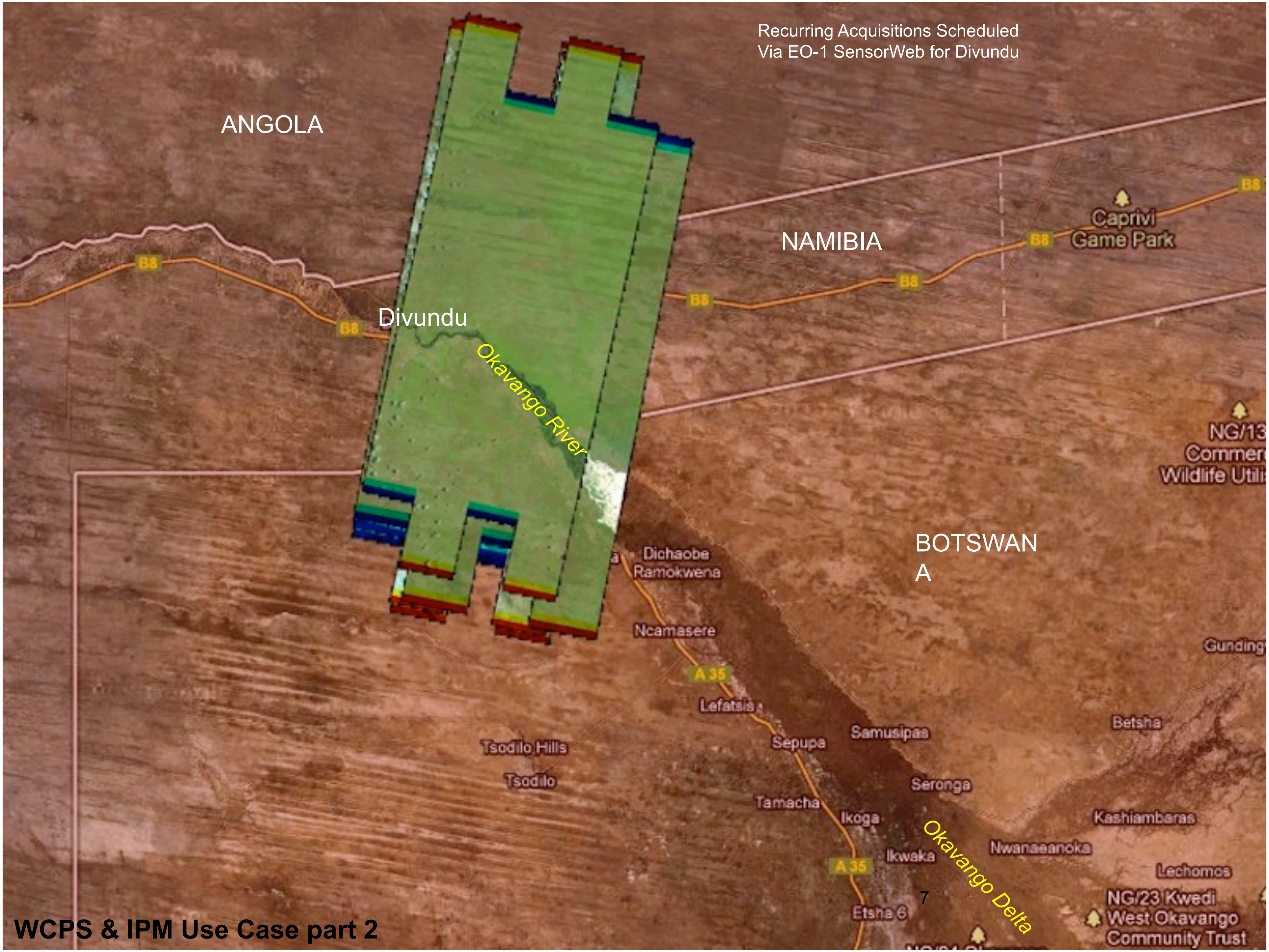
- For area upstream of the Okavango Delta
- Early 2012, establish ROI and standing request
- Baseline scenes acquired before onset
- Recurring acquisition per cloud predictions
- Earth Explorer reports new acquisitions
- WCPS cloud automatically ingests new data
- Select scenes and apply desired algorithm

5

Experimental IPM Quick Load/Quick Look Ops Con

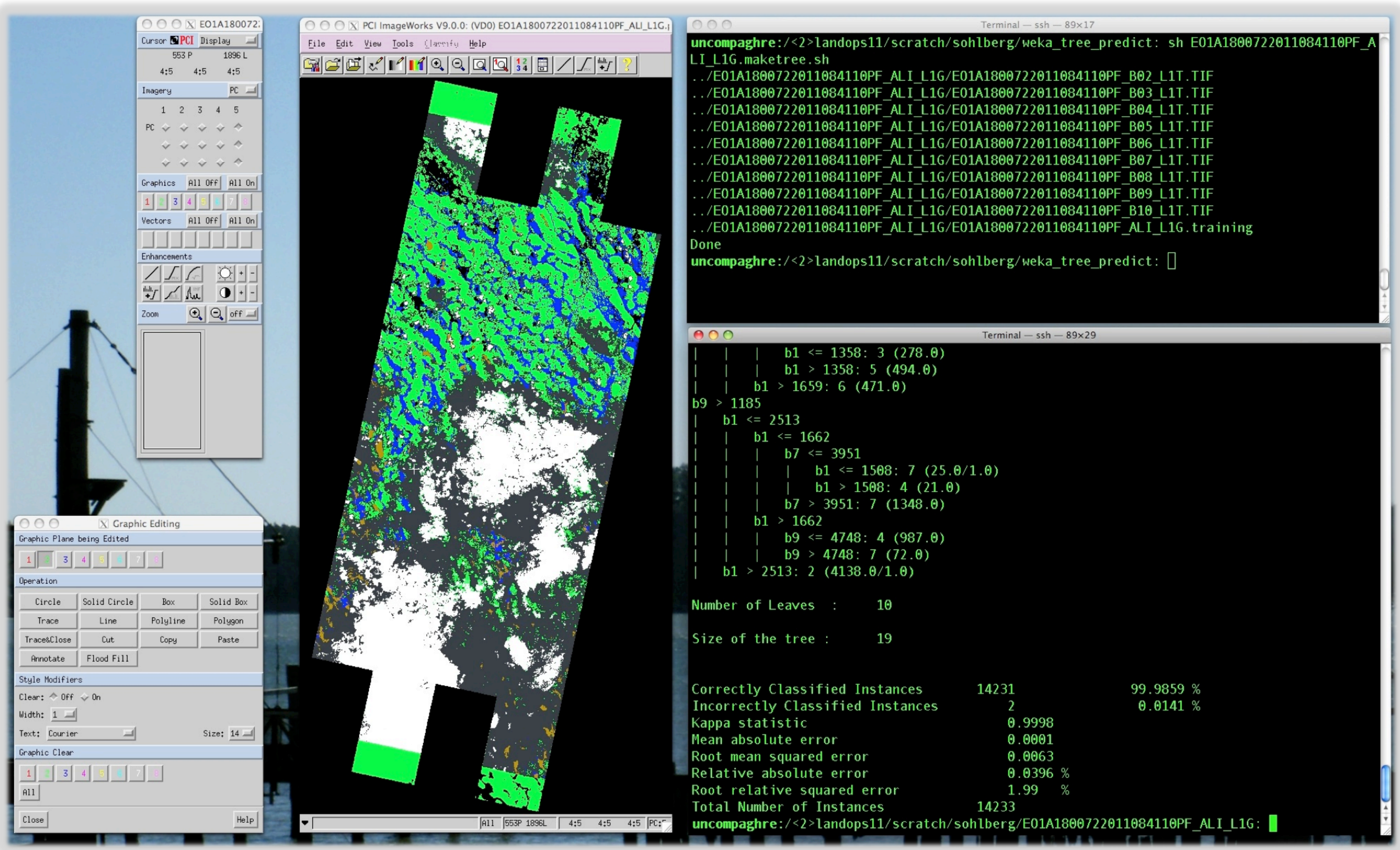
Web Coverage Processing Service (WCPS)-Client
Uploads to Various Environments



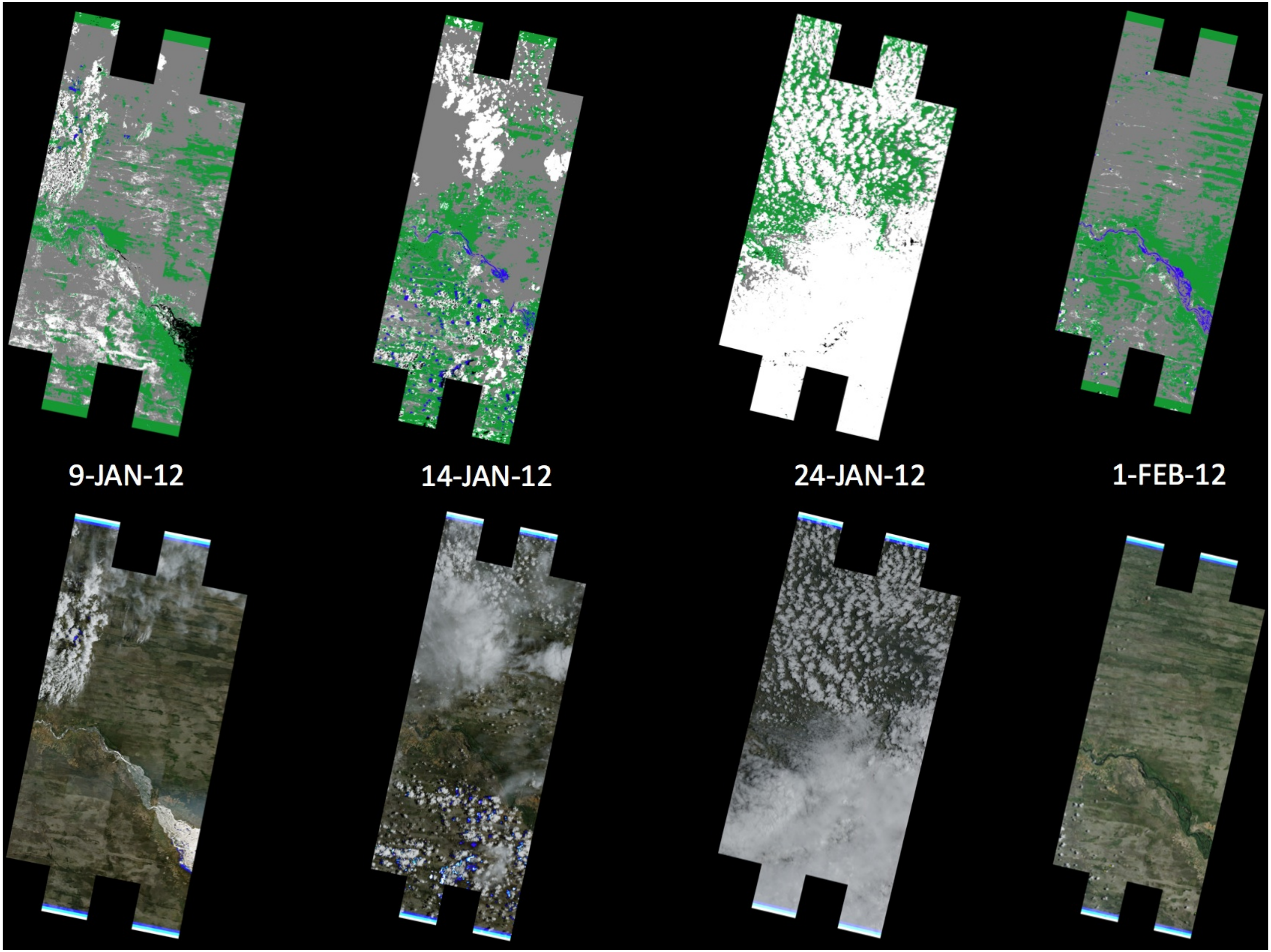


WCPS & IPM Use Case part 2

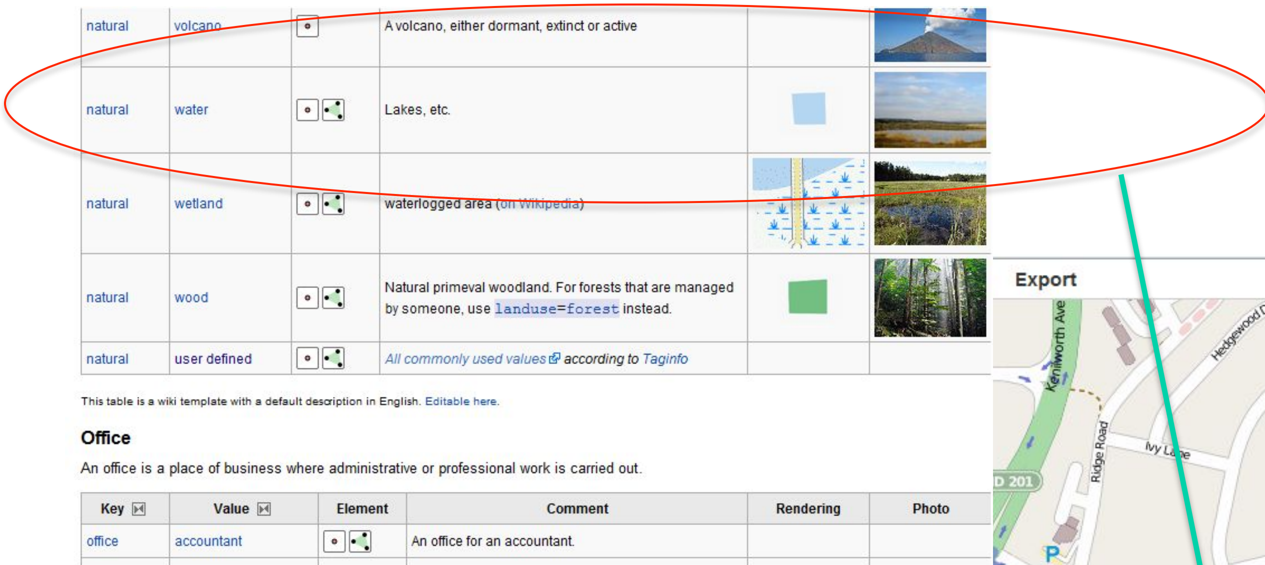
A model is developed using the Weka regression tree supervised classification algorithm.



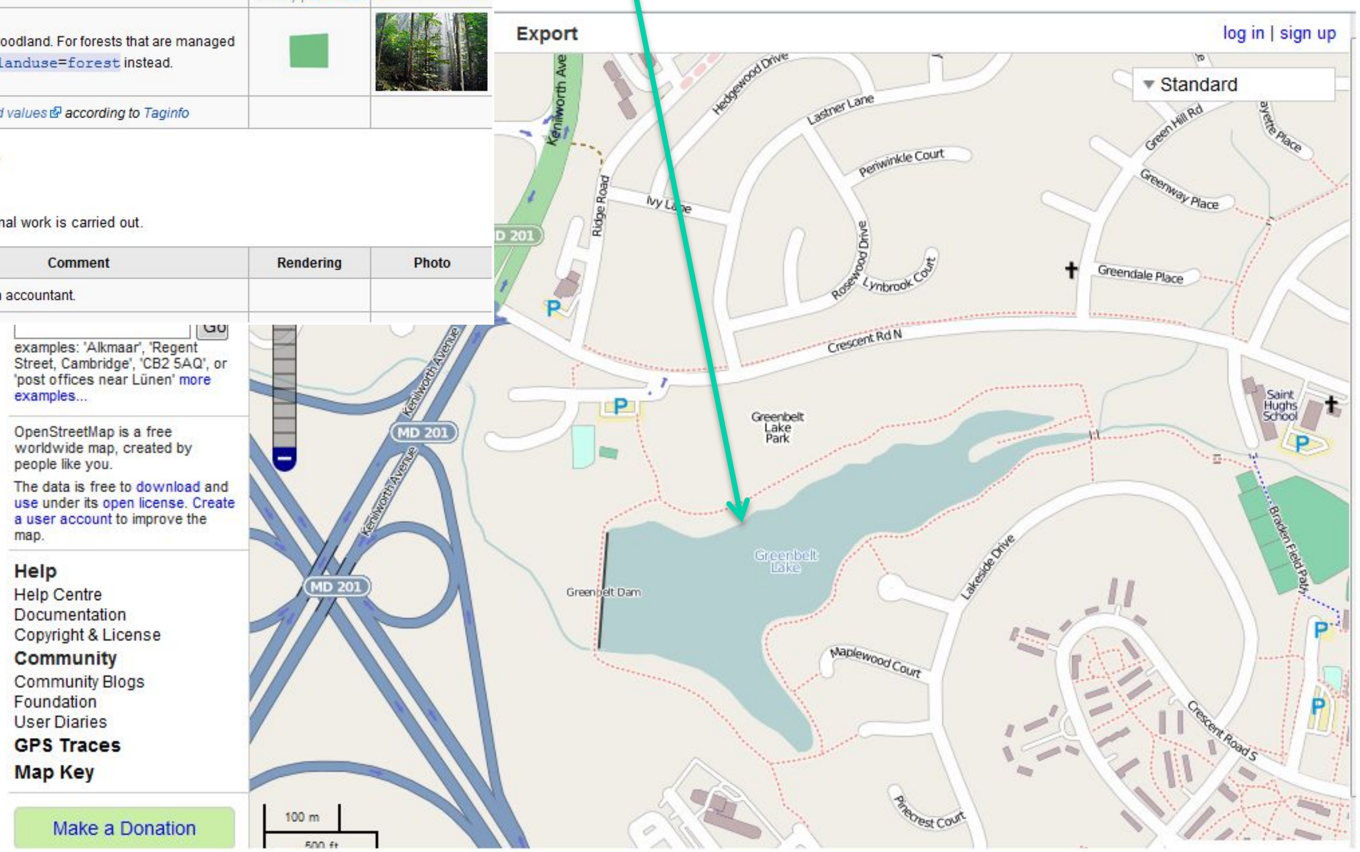
8


9-JAN-12
14-JAN-12
24-JAN-12
1-FEB-12

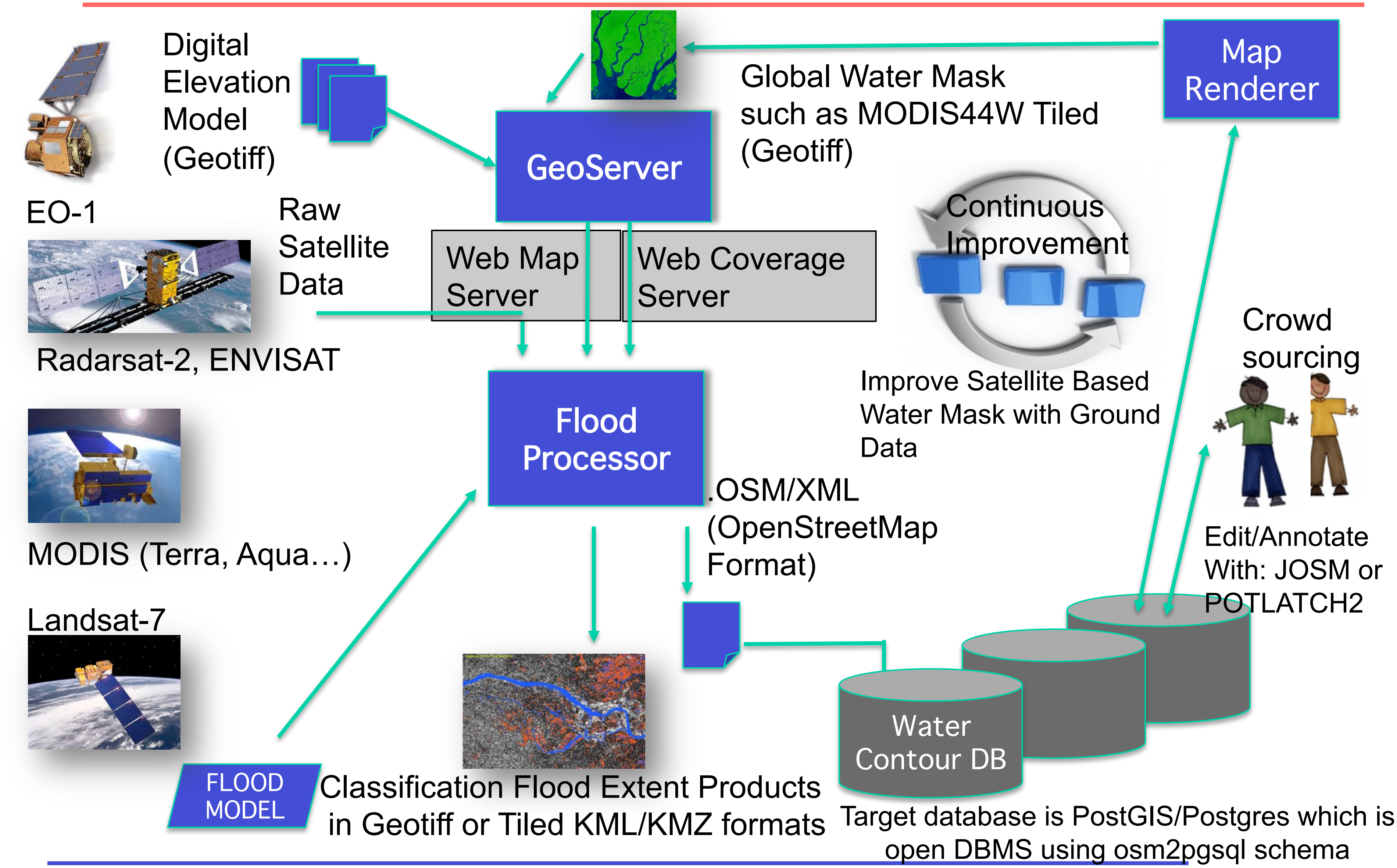
Use OpenStreetMap



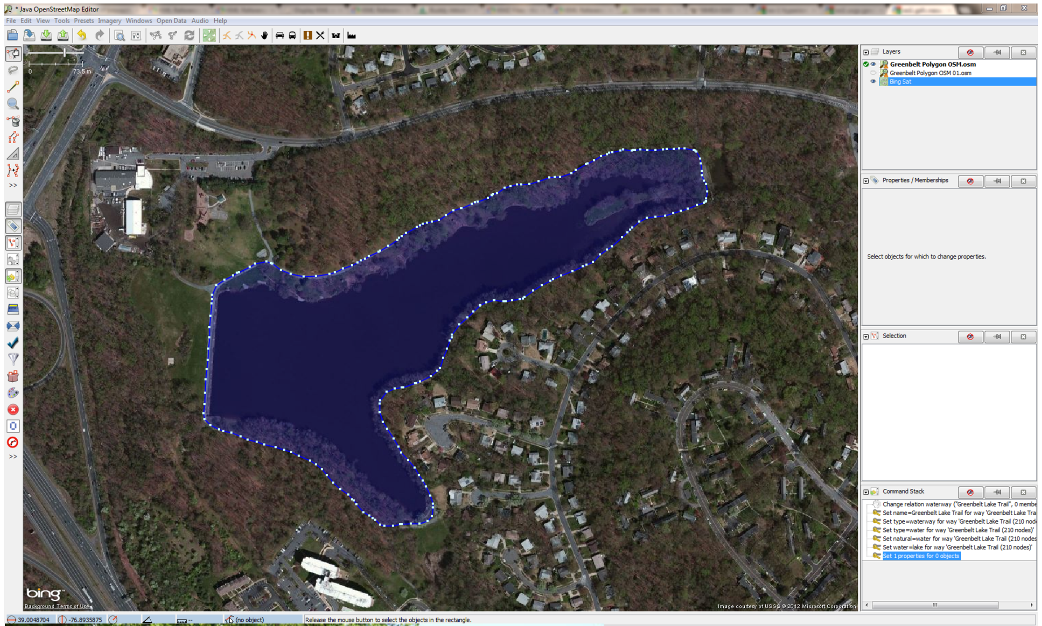
OpenStreetMap provides preset tags that enable map clients to automatically map polygon data as demonstrated here.



Vision of Generalized Water/Flood Architecture with Parallel OpenStreetMap (OSM) Format Output



Student Work – Java OpenStreetMap Editor Familiarization – GSFC Lake & Greenbelt Lake – Taking GPS Points to Lay Track on Map

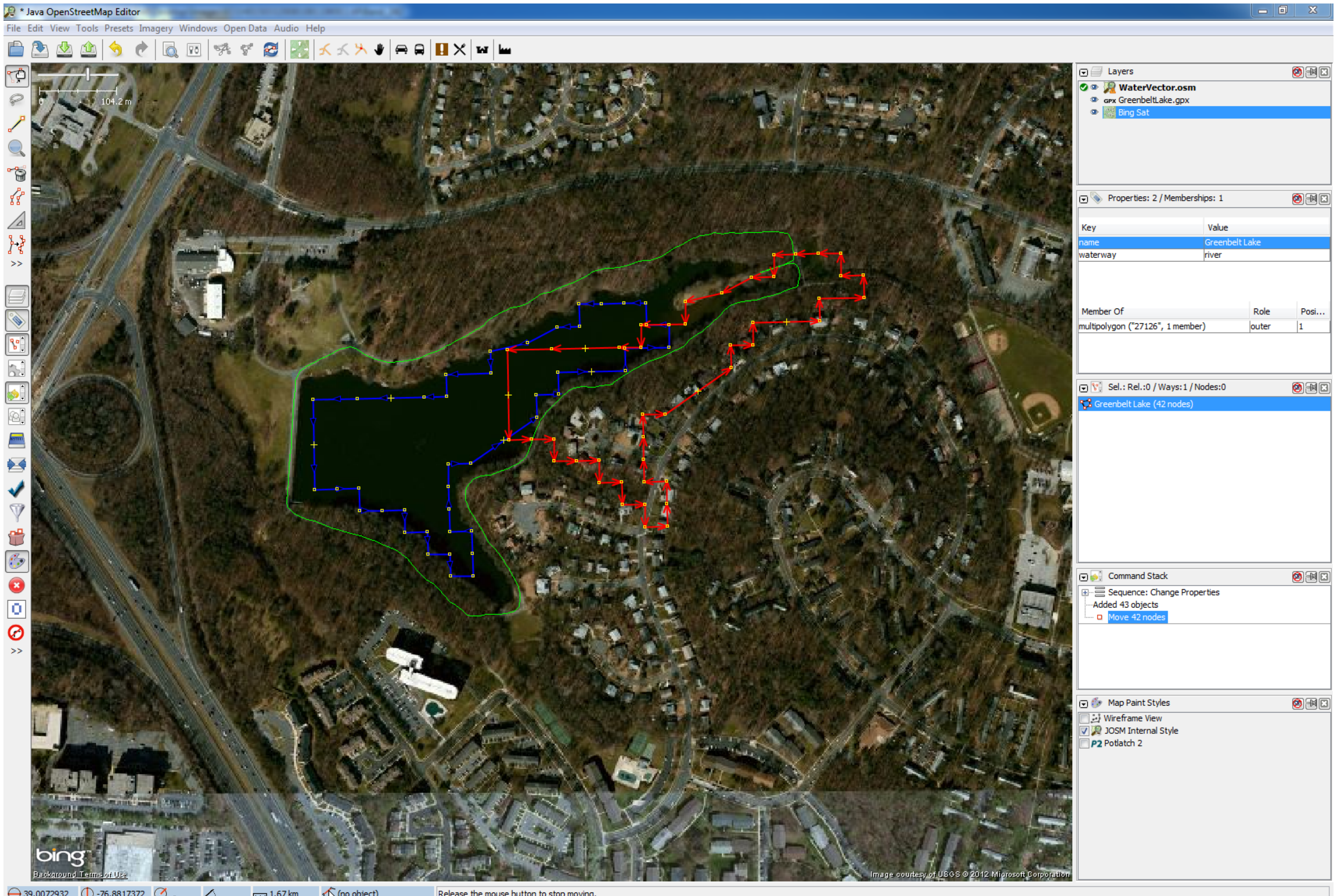


Joshua Bronston, Navajo Tech College GSFC Coop student – Pursuing Masters in Computer Engineering

Left: Neil Shah, Summer Intern, Univ. of Md College Park, major Aerospace Engineering, Middle: Chris Flatley, summer intern, Virginia Tech, major Computer Engineering

Left: Michael Mandl, Univ. of Md College Park, engineering student with Neil Shah and Chris Flatley

Experiment to Add Ground GPS Points, Add EO-1 ALI Water Detection Converted to Polygons and Begin to Edit in OSM



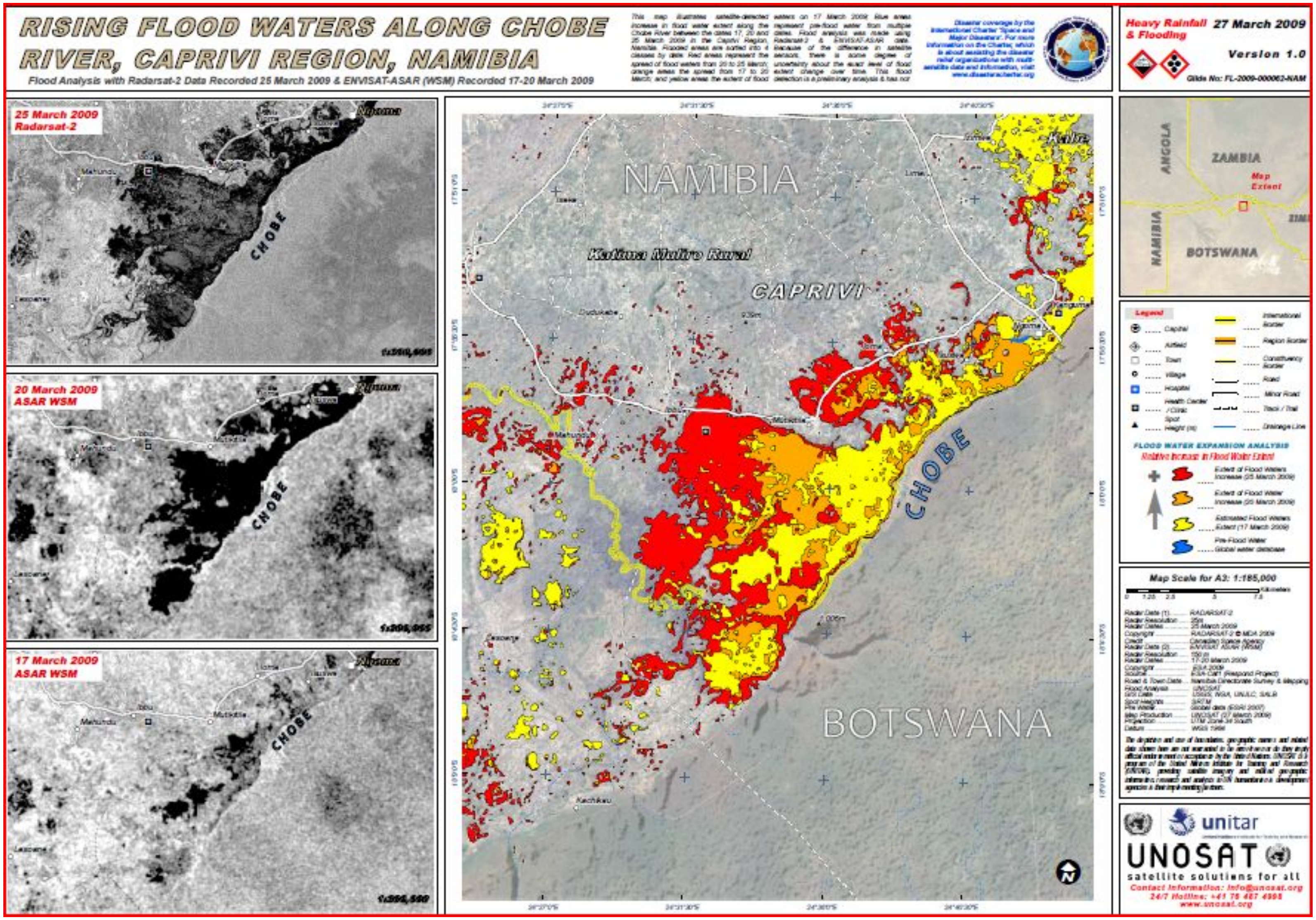
Green track is GPS track from walk around lake
Red track is converted polygon representing water contour from EO-1 ALI (known approx. 300 meter offset)
Blue track is use of JOSM to move satellite derived polygon using JOSM editing capability

Automatic Time Series for Floods

Potential flood progression created with Java OpenStreetMap (JOSM).

- OpenStreetMap is a free tool that allows crowd-sourced data to be combined with SensorWeb data.
- This makes it easier to create more comprehensive data products.

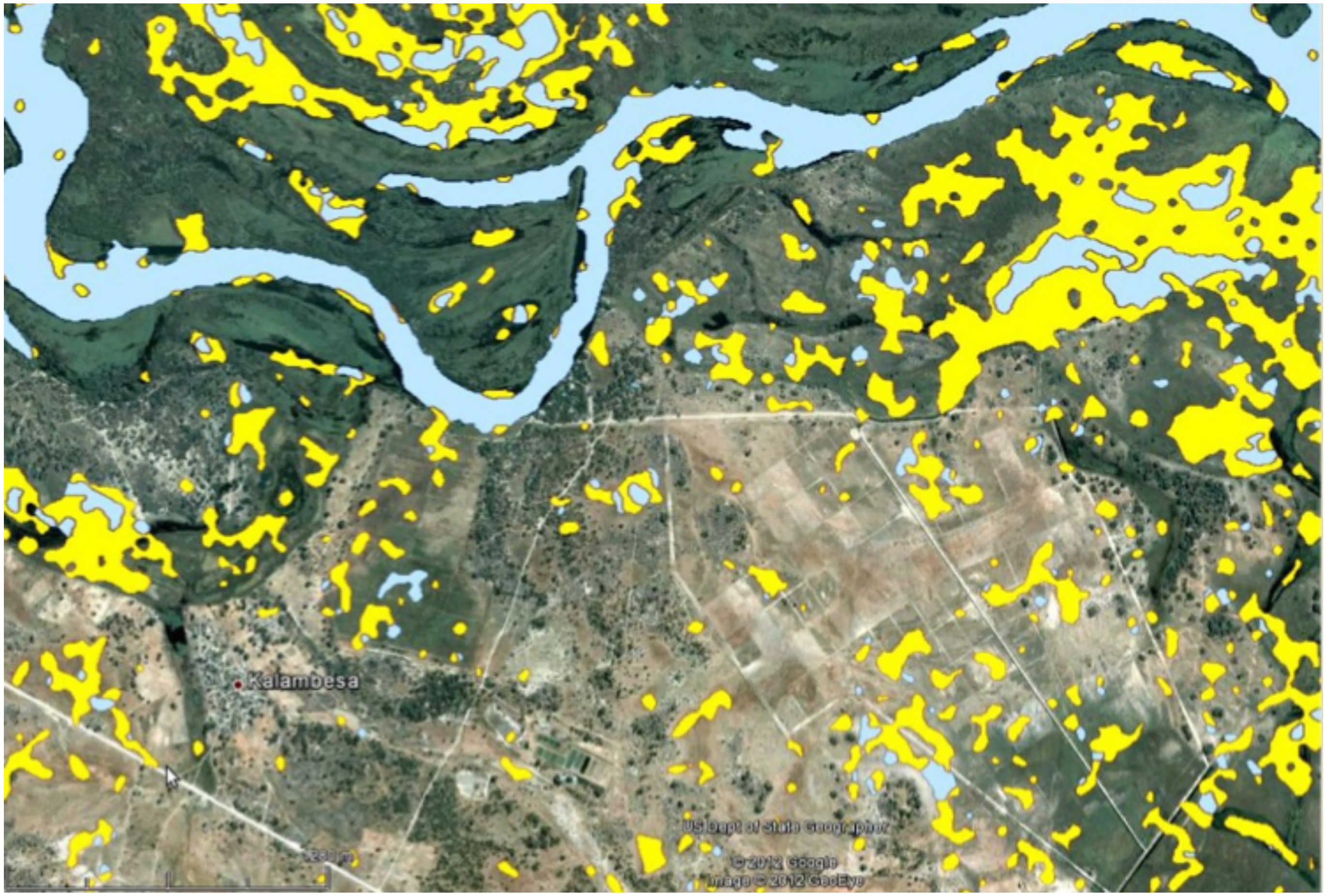
Future Goal is to Automate Creation of Time Series Differential Map, Below is Example Manually Created in 2009 by Unosat



Use OpenStreetMap

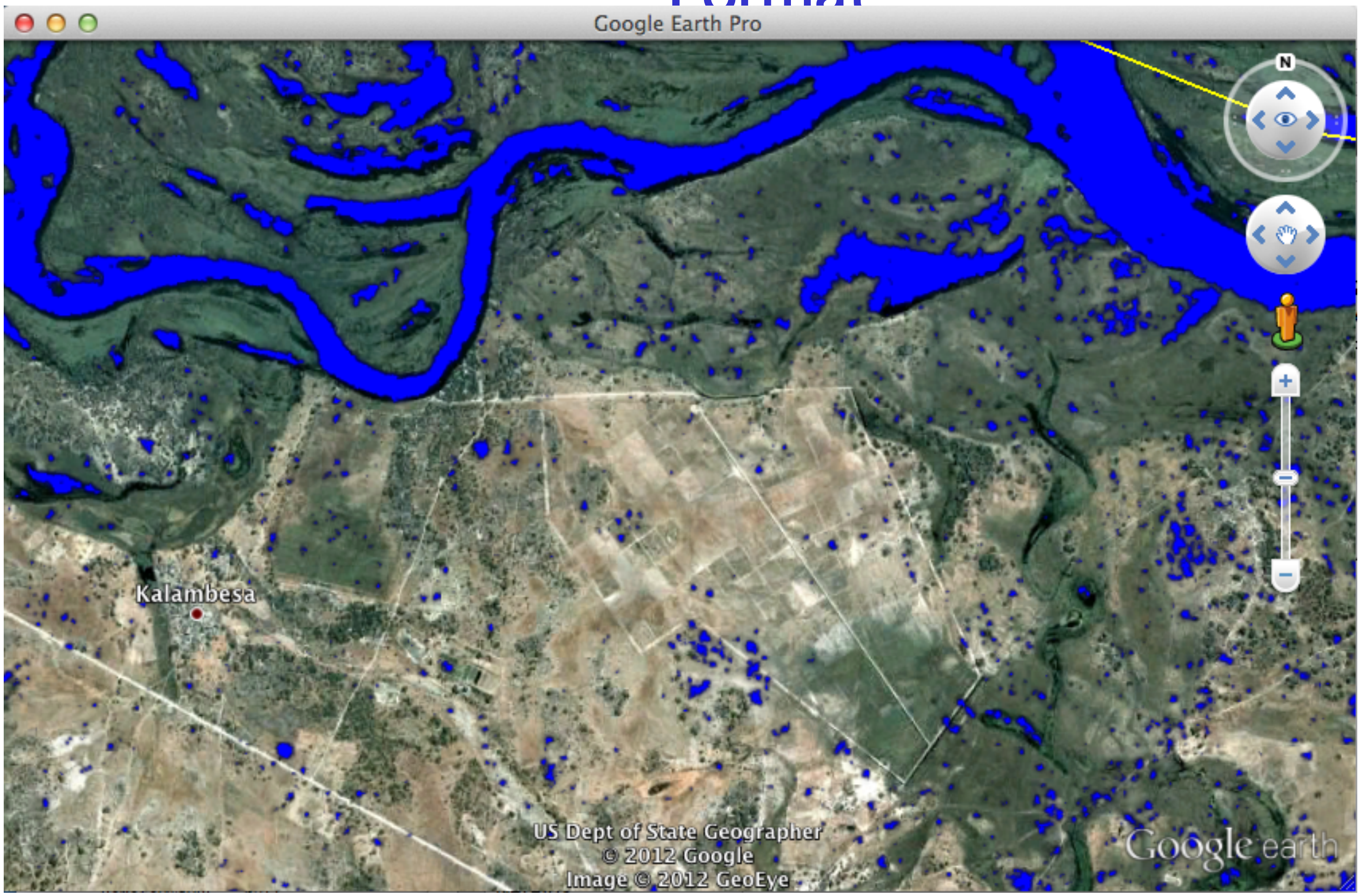
1. Use Planet.osm to store and ever improving base water mask with contributions from many sources
   - Use commonly used tags to maximize interoperability
   - Use standard tags to enable use of standard map clients
2. Use combination of standard and augmented tags to enable calibration and validation of satellite images
   - Define tags and terms needed for use by hydrologists (e.g. error locations of water)
   - Query database and customize output display
3. Use heterogeneous data base of water contours to query data base and create customized time series of water progression of floods from multiple sources

Radarsat Processing Into Flood Extent



Kavango River in Namibia. Radarsat image processed manually by MacDonald Detweiller and Associates (MDA) as a PDF shape file (blue: open water; yellow: inundated), derived from the image processing applied to the Feb. 17, 2012 RADARSAT-2 image, converted into KML format and displayed in Google Earth.

Prototype Automated Radarsat Water Extent (without Inundation Differentiation) in Tiled Geotiff Format

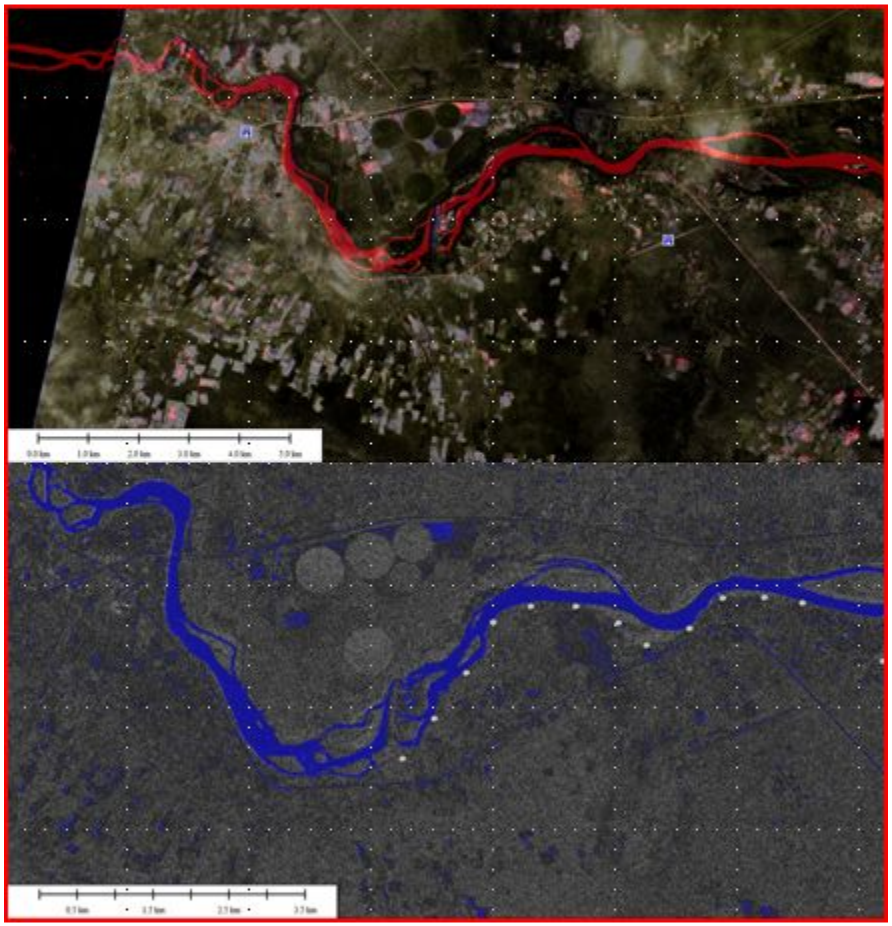


Same Kavango image, Feb 17, 2012 in Namibia but processed with our automatic Radarsat processor algorithm with tiled output running on laptop. Goal is to run on Matsu and Joyent clouds to make it a “do-it-yourself” process.

Next Steps that are in process; Convert Geotiffs to Polygons with OpenStreetmap Tags

- Original Geotiff file was 1.2 Gbytes
- Converted OpenStreetmap file was 2.4 Gbytes
- Took 24 hours to process
- Need streamlined methods to make this happen
- Also need to identify tags to use when this conversion is done to make it useful for hydrologists

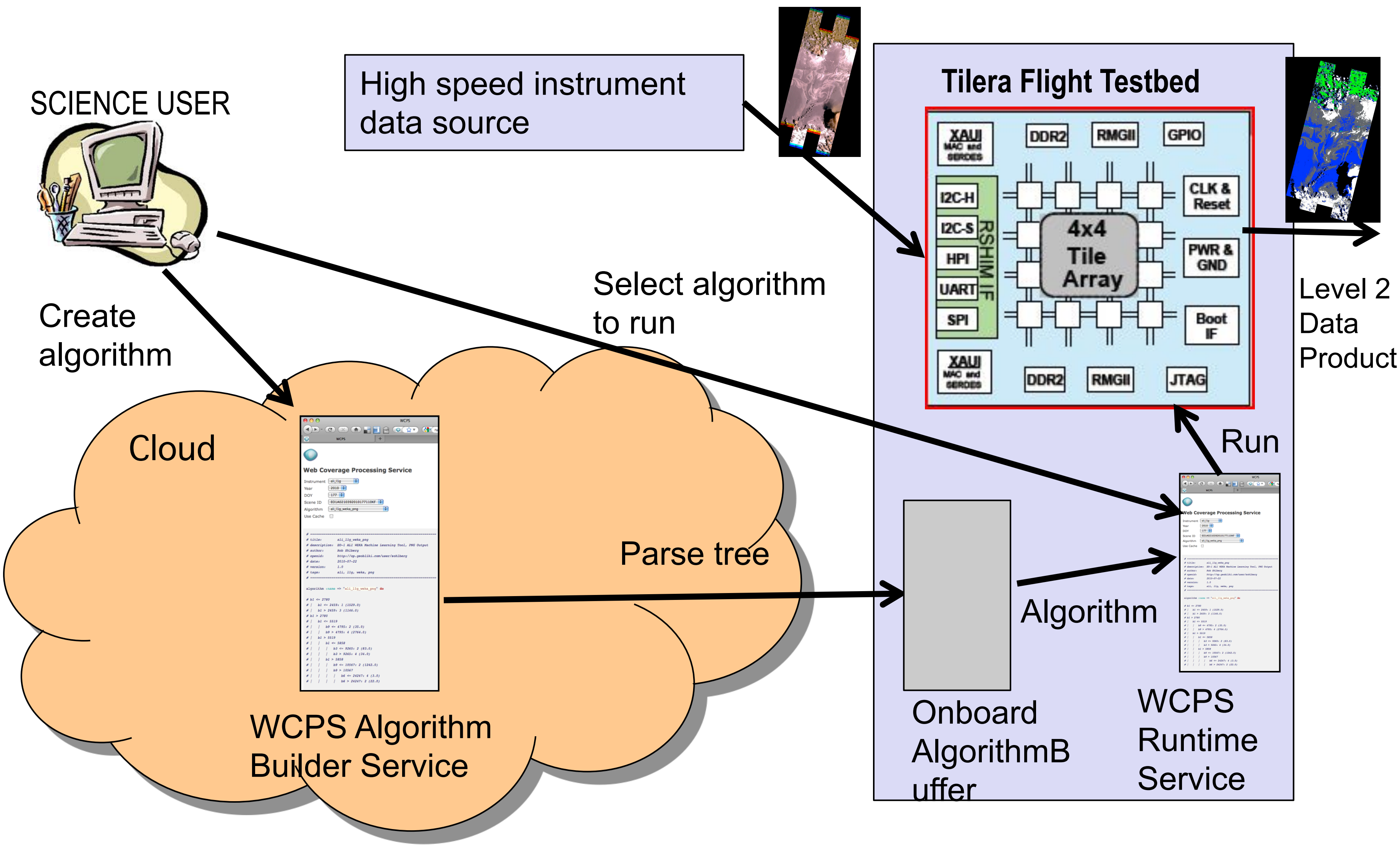
Repeat Process with Namibia Data Gathered January 2012 Radarsat, EO-1 and Ground GPS (late summer 2012)



McCloud Katjizeu (orange) Dept. of Hydrology compares GPS readings of control point with U. Namibia students for mapping exercise.

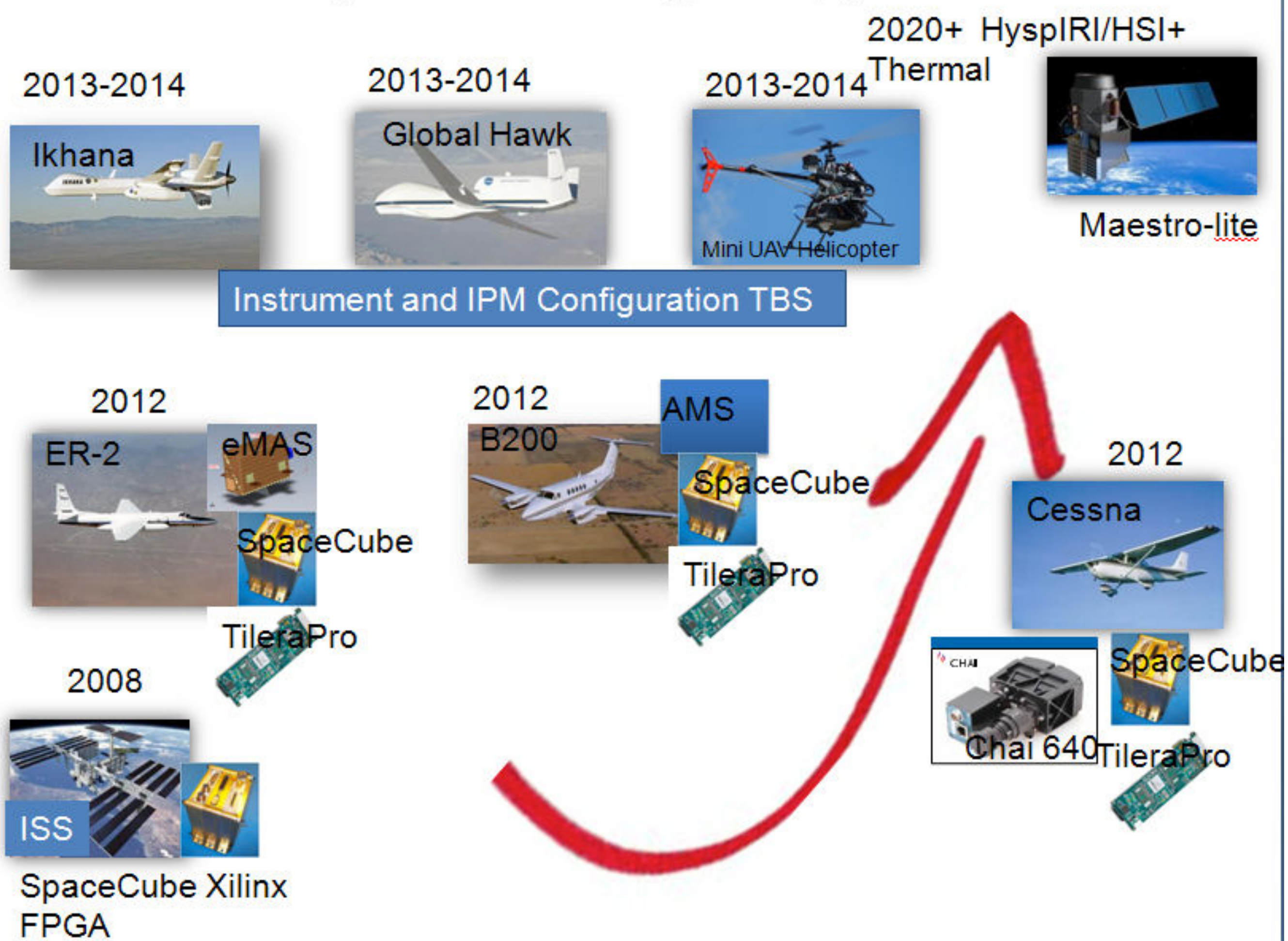
Georeferenced photos enable Rob Sohlberg/UMD to train classifier algorithm to detect presence of water in grassy marsh lands from satellite data.

Onboard Processing Controlled by SensorWeb Software



21

Roadmap for IPM Flight Opportunities



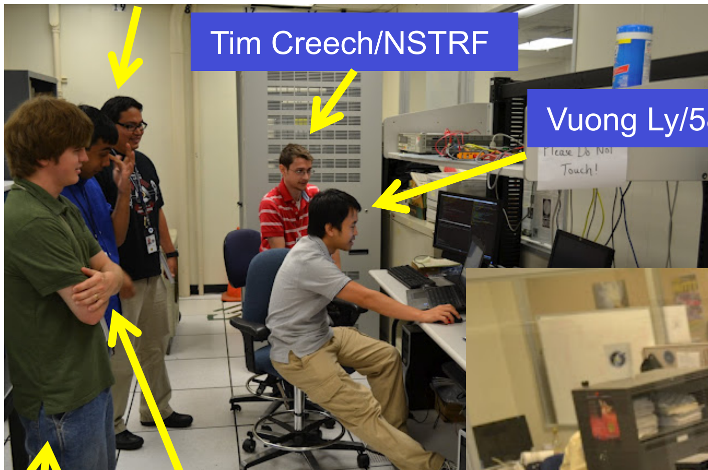
22

Intelligent Payload Module (IPM) Prototype without Box

Joshua Bronston/581 coop

Tim Creech/NSTRF

Vuong Ly/583

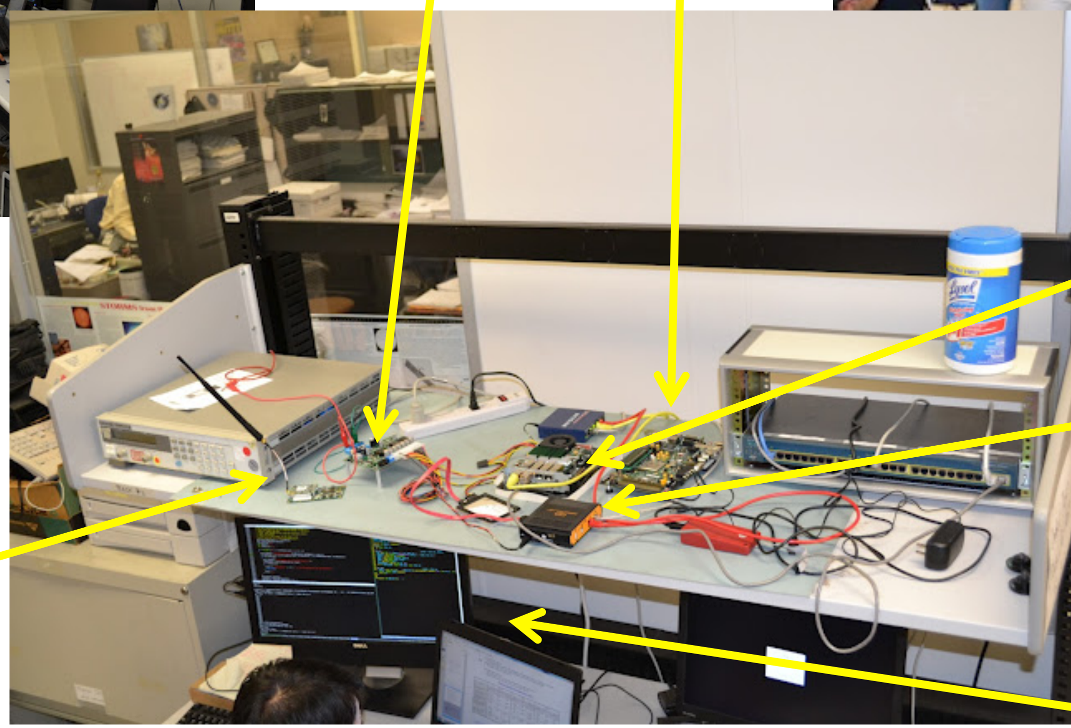


Neil Shah/581 summer intern

Matt Handy/583

SpaceCube board

Power supply



Tilera board

Dream plug (processor)

Freewave Radio Transceiver & Antenna

Solid state disk

23

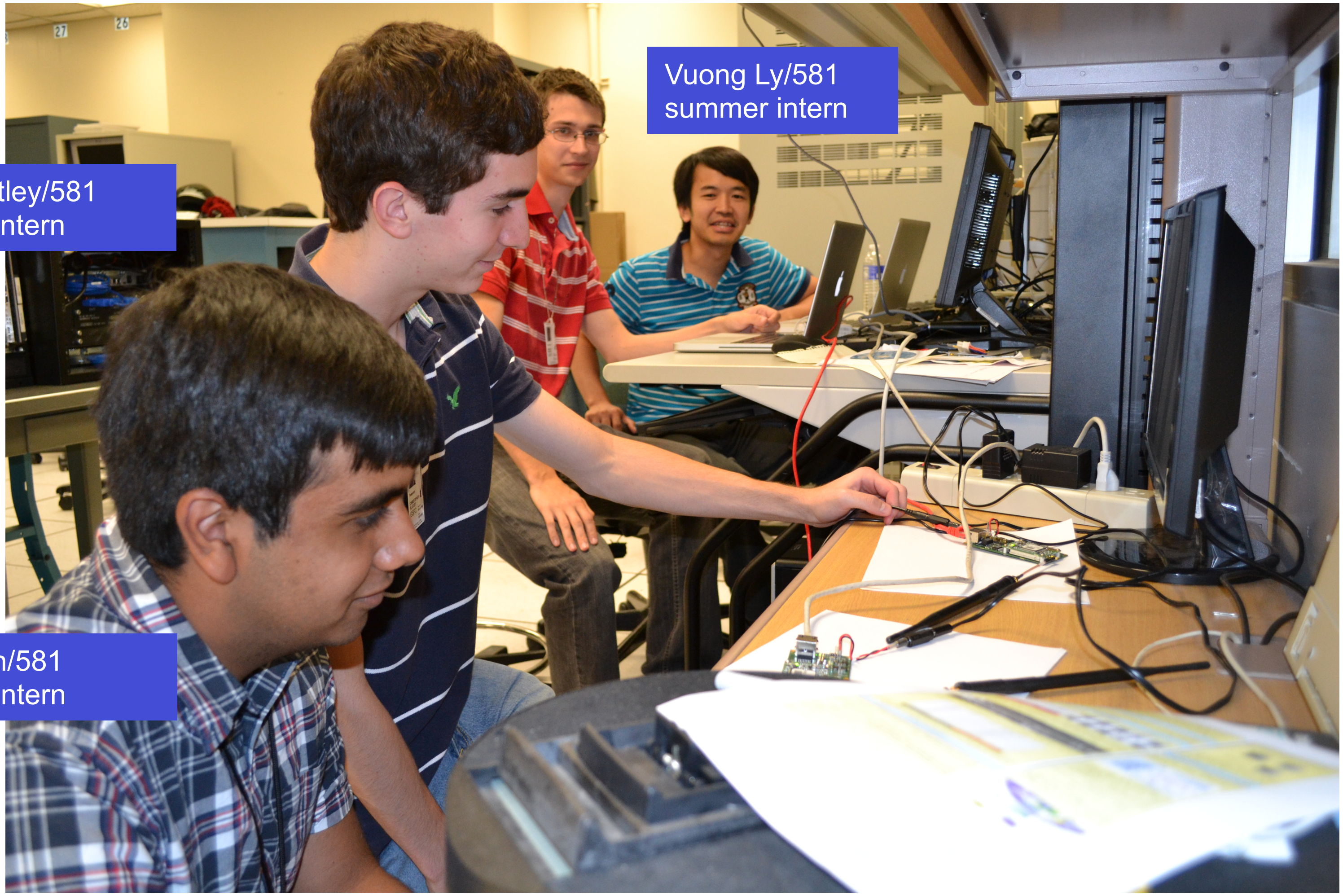
Testing Freewave Radio

Tim Creech/NSTRF

Vuong Ly/581
summer intern

Chris Flatley/581
summer intern

Neil Shah/581
summer intern



24

IPM Testbed with Tilera Acting as Proxy for Maestro & Maestro-lite Board (building 23)

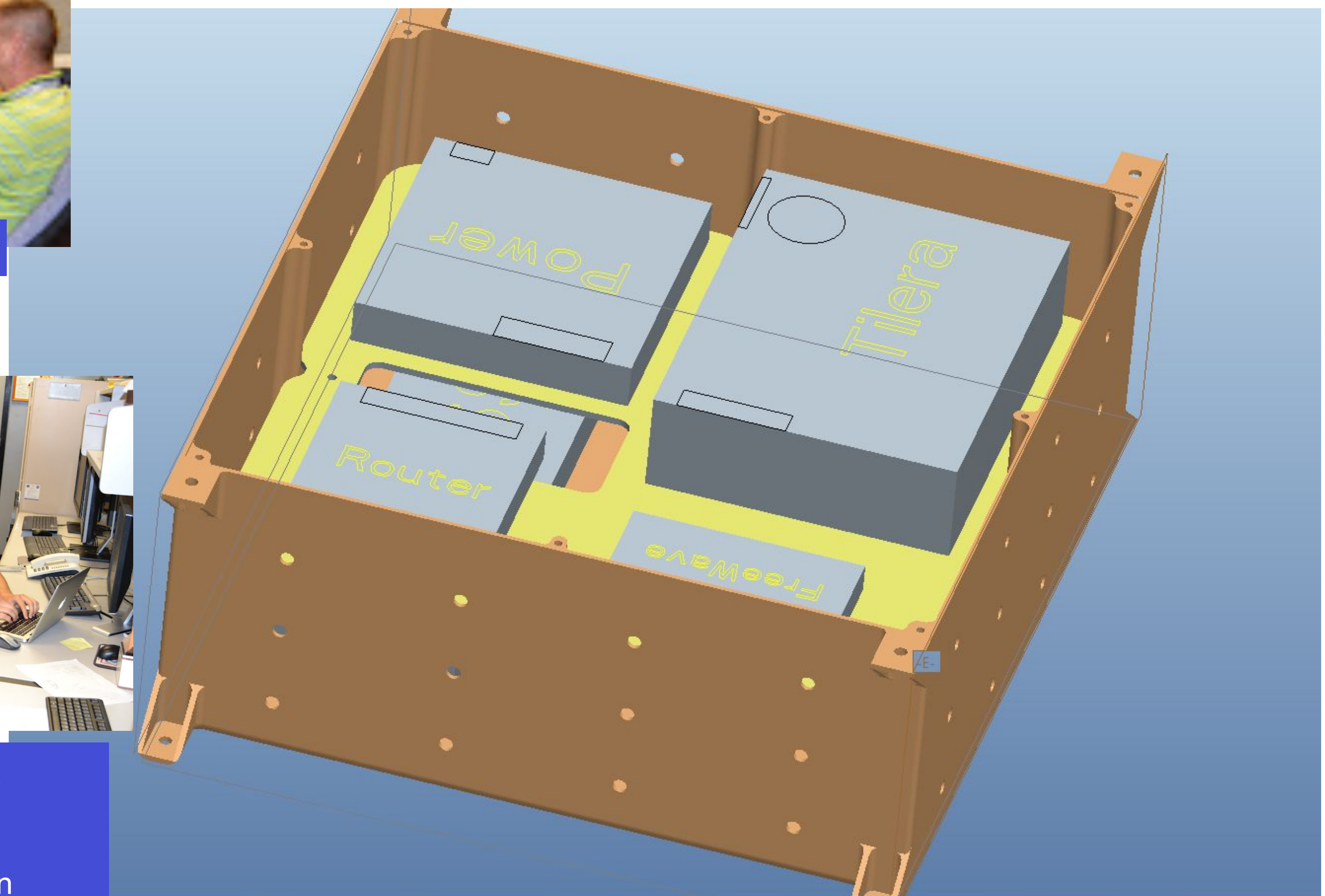
Above: Vuong Ly/583 (Ground System SW Branch) standing front of IPM testbed holding a SpaceCube board.
Right: Tawanda Jacobs/582 (Flight SW Branch) working on integration of cFE onto IPM testbed.

25

Designed Box with Pro-E to House IPM Components

Mike Flick/SES

Mike Mandl/UMCP Student
Mike Flick
Neil Shah/Summer Intern
Chris Flatley/Summer Intern



26

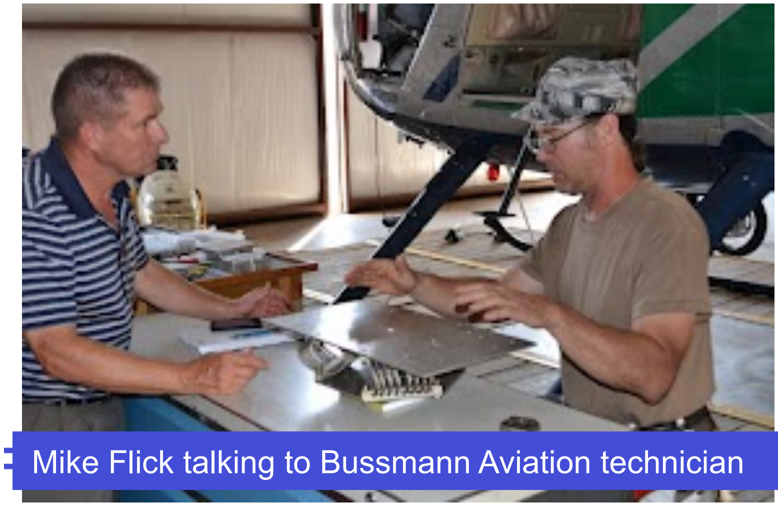
Plan to Mount IPM to Helicopter for Tests Under AIST ESTO Research Grant

Vibration isolation mount for helicopter

Mike Mandl/UMCP student

Chris Flatley/581 Summer intern

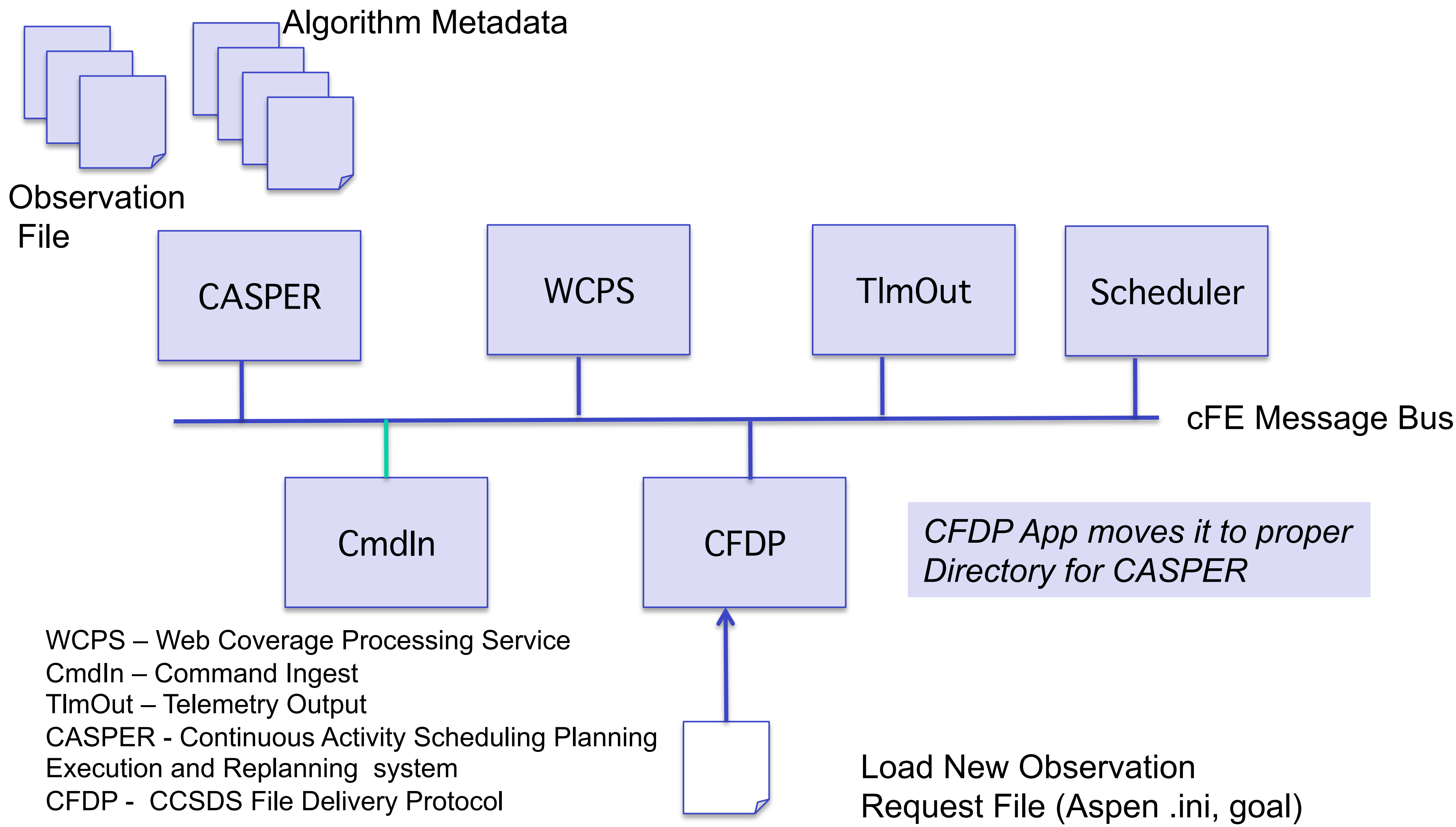
Joshua Bronston/581



Mike Flick talking to Bussmann Aviation technician

Steve Bussmann/Bussmann Aviation

Flight Architecture

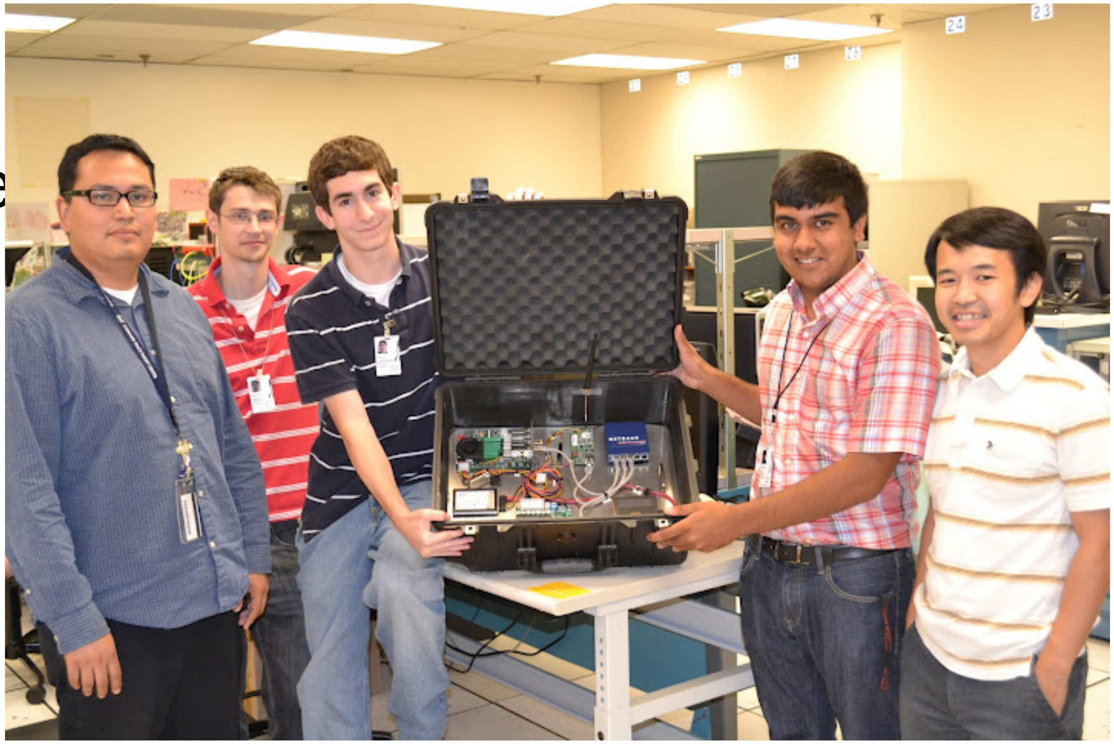


WCPS – Web Coverage Processing Service
CmdIn – Command Ingest
TlmOut – Telemetry Output
CASPER - Continuous Activity Scheduling Planning Execution and Replanning system
CFDP - CCSDS File Delivery Protocol

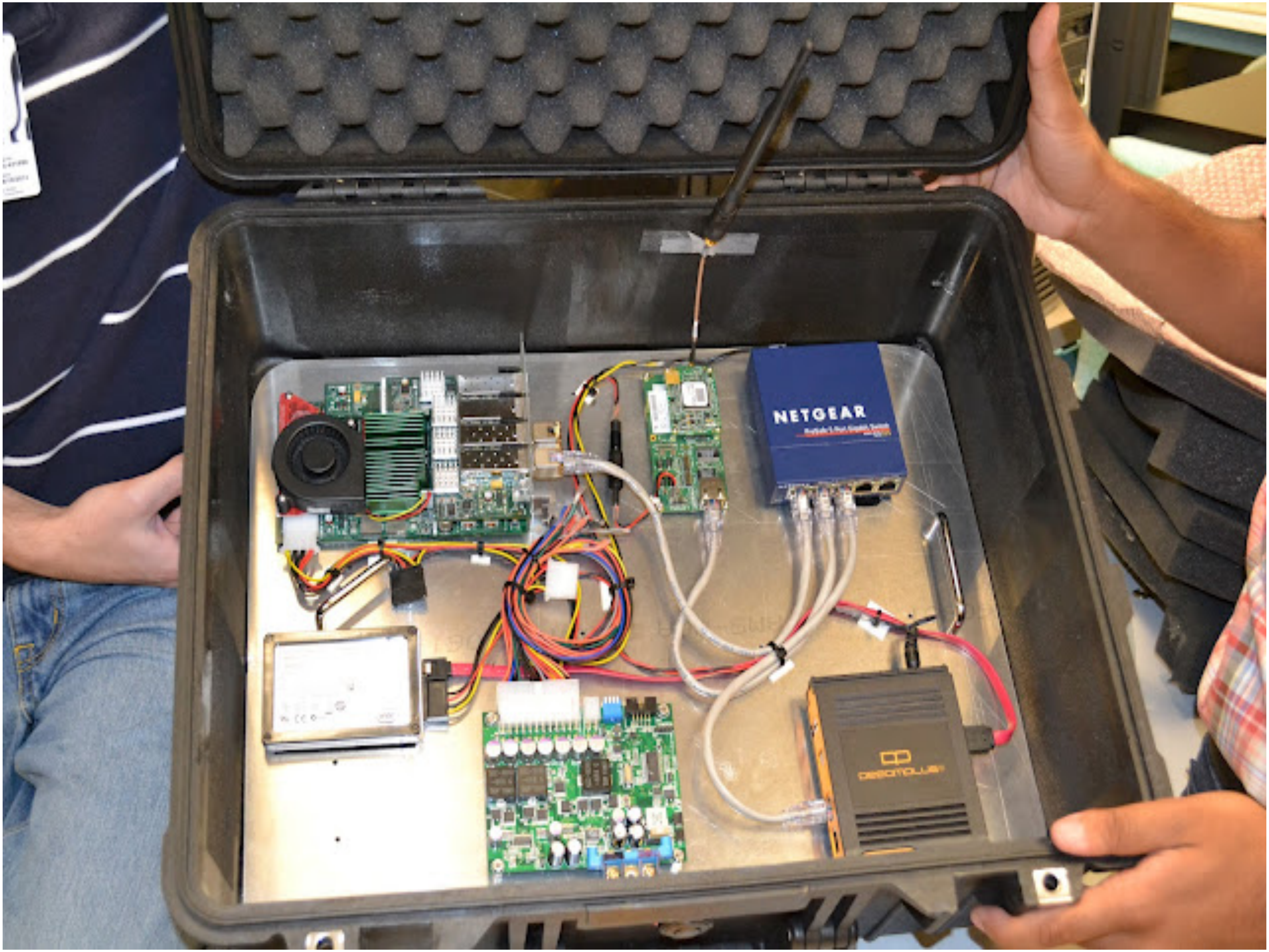
28

Components

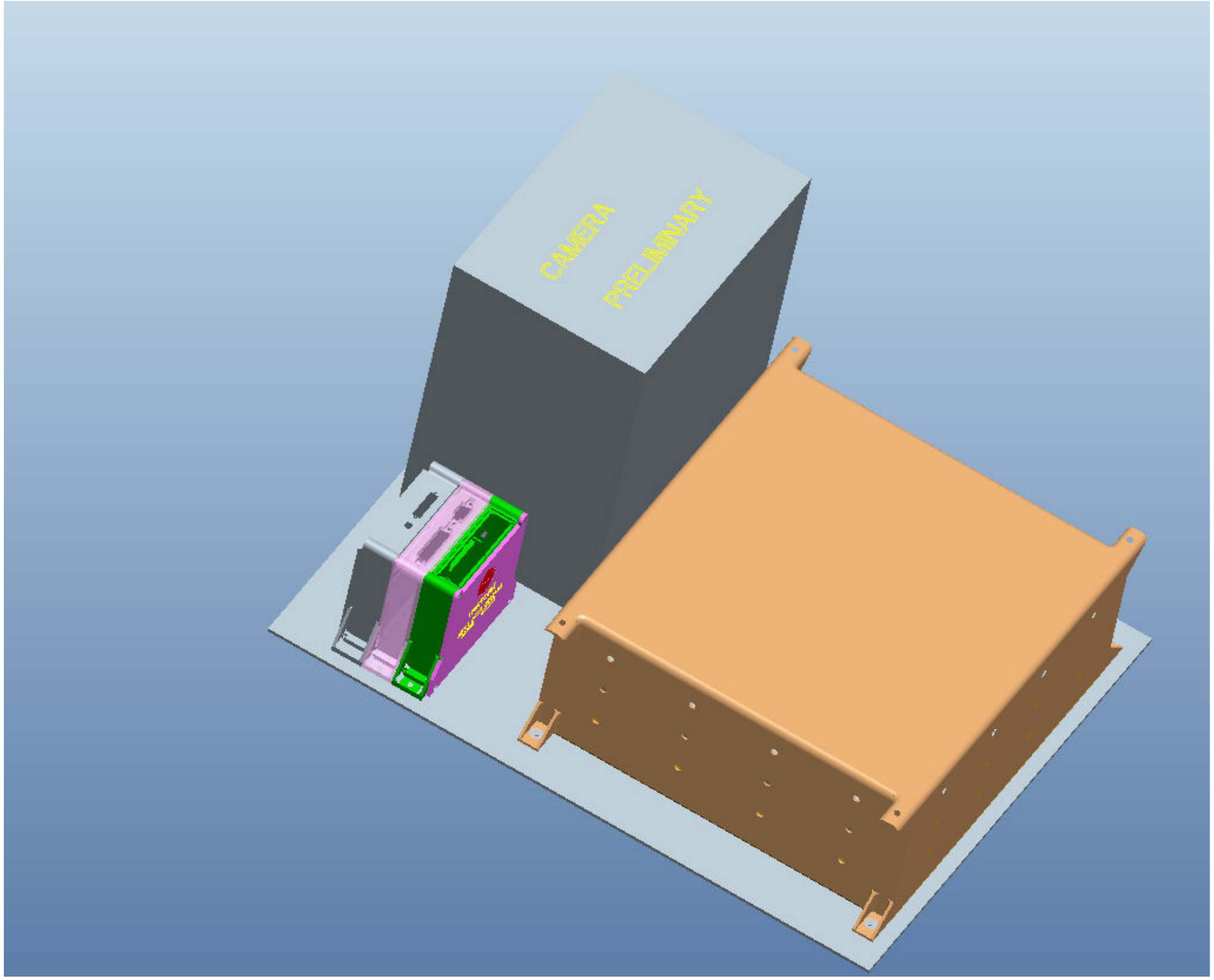
- Tilera TILEPro
- SpaceCube
- DreamPlug
- Solid State Hard Drive
- Power Board
- Router
- FreeWave Radio



The IPM team (from left to right): Josh Bronston (581), Tim Creech (UMD), Chris Flatley, Neil Shah, Vuong Ly (583)

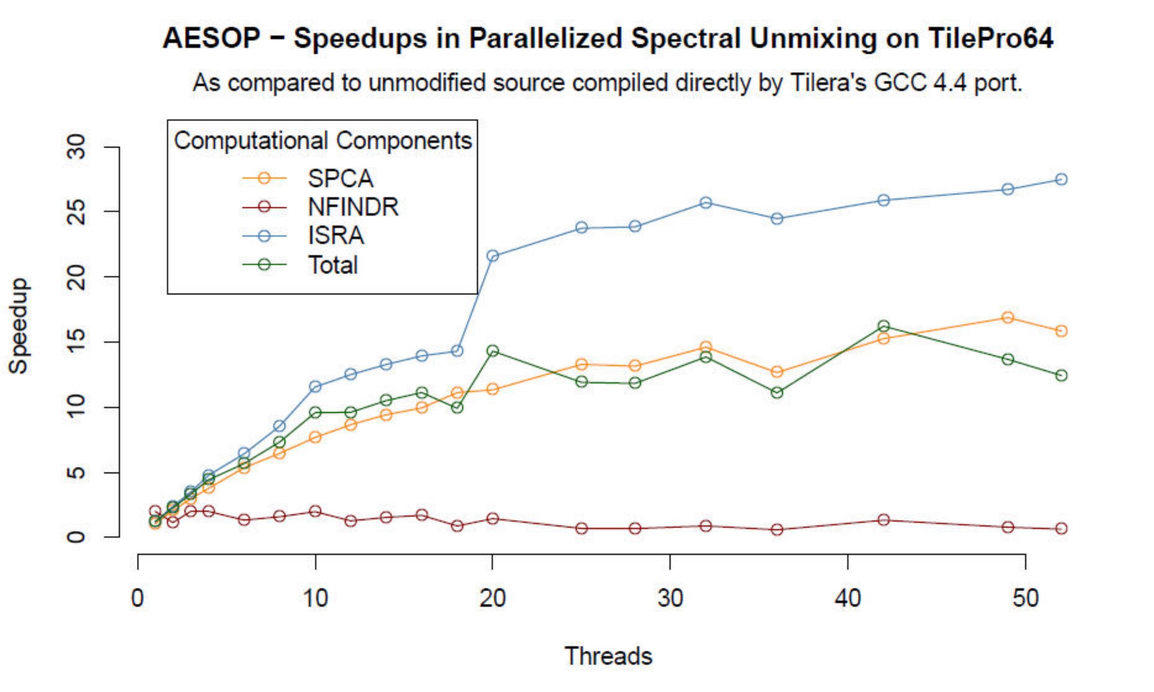

NETGEAR

IPM Assembly with Tilera Box, SpaceCube Box and Chai640 Instrument Box



31

Auto-parallelizing Using AESOP on Spectral Unmixing Algorithm



- Speedups on the TilePro64 obtained entirely automatically
  - ✓Tool chain consists of ~1000 lines of C code
  - ✓Written by a NASA collaborator (Antonio Plaza) for x86 with no special knowledge of or intent for AESOP or TilePro64 hardware
  - ✓ Speedups shown are for TilePro64, used as a proxy for Maestro which was not yet available

✓

32

Conclusion

- SensorWebs provide opportunity to gather sensor data, especially, satellite sensor data, cost-effectively, rapidly and on a “do-it-yourself” basis
- SensorWebs leverage interoperable open standards
- SensorWebs are good tools when rapid decision support is